

**WORKMAN
NYDEGGER
& SEELEY**ATTORNEYS AT LAW
A PROFESSIONAL CORPORATION

1000 EAGLE GATE TOWER

60 EAST SOUTH TEMPLE

SALT LAKE CITY, UTAH 84111

TELEPHONE (801) 533-9800

FACSIMILE (801) 328-1707

RICK D. NYDEGGER
DAVID O. SEELEY
BRENT P. LORIMER
THOMAS R. VUKSINICK
LARRY R. LAYCOCK
JONATHAN W. RICHARDS
DAVID R. WRIGHT
JOHN C. STRINGHAM
BRADLEY K. DESANDRO
JOHN M. GUYNN
CHARLES L. ROBERTS
GREGORY M. TAYLOR
DANA L. TANGREN
KEVIN B. LAURENCE
ERIC L. MASCHOFF
CHARLES J. VEVERKA
ROBYN L. PHILLIPS
RICHARD C. GILMORE†DAVID B. DELLENBACH
KEVIN K. JOHANSON
L. DAVID GRIFFIN
R. BURNS ISRAELSEN
DAVID R. TODD
FRASER D. ROY
CARL T. REED
JESÚS JUANÓS I TIMONEDA, Ph.D.
STEPHEN D. PRODNUK, Ph.D.
R. PARRISH FREEMAN, Jr.
PETER F. MALEN, Jr.
ADRIAN J. LEE
KYLE H. FLINDT

†ADMITTED ONLY IN CALIFORNIA

H. ROSS WORKMAN
OF COUNSELPATENT
TRADEMARK
COPYRIGHT
TRADE SECRETS
UNFAIR COMPETITION
LICENSING
COMPLEX LITIGATIONMAILING ADDRESS:
P.O. BOX 45862
SALT LAKE CITY, UT 84145

INTERNET

HOME PAGE: <http://www.wnspat.com>GENERAL E-MAIL: info@wnspat.com

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Box: PATENT APPLICATION
Assistant Commissioner for Patents
Washington, DC 20231jc678 U.S. PTO
09/527137
03/16/00TRANSMITTAL FOR PATENT APPLICATION

Sir:

Transmitted herewith for filing under 37 C.F.R. § 1.53(b) is a United States patent application entitled APPLICATION PROGRAM INTERFACES FOR ELECTRONIC PROGRAM GUIDE DATA SERVICES in the name of the following inventors:

Inventor:	Michael E. Pietraszak
Inventor:	Robert M. Fries
Inventor:	Robert S.T. Gibson
Inventor:	Jonathan A. Wray

Enclosed are the following:

- X A specification, claims, abstract, and cover page in total comprising forty-one (41) pages.
- X Three (3) sheets of drawings.
- X A Certificate of Express Mailing by "Express Mail" certifying a filing date by use of Express Mail Label No. EL544448887US.
- X No fees are enclosed.
- X No Declaration, Power of Attorney, and Petition is enclosed.

03/16/00
jc678 U.S. PTO

09527137-031600

X Priority to United States Provisional Application Serial No. 60/125,062 filed on March 18, 1999, is claimed under 35 U.S.C. § 119.

 A certified copy of foreign priority Application Serial No. _____.

 An Associate Power of Attorney.

Please address all future correspondence in connection with the above-identified patent application to the attention of the following:

RICK D. NYDEGGER
WORKMAN, NYDEGGER & SEELEY
1000 Eagle Gate Tower
60 East South Temple
Salt Lake City, Utah 84111
Telephone: (801) 533-9800
Facsimile: (801) 328-1707

Dated this 16th day of March 2000.

Respectfully submitted,



ADRIAN J. LEE
Attorney for Applicant
Registration No. 42,785

WORKMAN, NYDEGGER & SEELEY
1000 Eagle Gate Tower
60 East South Temple
Salt Lake City, Utah 84111
Telephone: (801) 533-9800
Facsimile: (801) 328-1707

AJL:CTR:dfw

G:\DATA\WPDOCSRN\MICROSOFT\OTHERDOC\0316pat_trx_57_2.doc

009760-0316-0000

UNITED STATES PATENT APPLICATION

of

Michael E. Pietraszak

Robert M. Fries

Robert S.T. Gibson

and

Jonathan A. Wray

for

**APPLICATION PROGRAM INTERFACES FOR
ELECTRONIC PROGRAM GUIDE DATA SERVICES**

WORKMAN, NYDEGGER & SEELEY

A PROFESSIONAL CORPORATION

ATTORNEYS AT LAW

1000 EAGLE GATE TOWER

60 EAST SOUTH TEMPLE

SALT LAKE CITY, UTAH 84111

OFFICE OF THE CLERK

"Express Mail" Mailing Label No.: EL544448887US

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

- Patent Application in the names of Michael E. Pietraszak, Robert M. Fries, Robert S.T. Gibson, and Jonathan A. Wray for APPLICATION PROGRAM INTERFACES FOR ELECTRONIC PROGRAM GUIDE DATA SERVICES (41 pages)
- Drawings (3 sheets)
- Transmittal Letter (3 pages)
- Certificate of Express Mailing
- Postcard

Respectfully submitted,

ADRIAN J. LEE
Attorney for Applicant
Registration No. 42,785

Ajl:CTR:dfw

BACKGROUND OF THE INVENTION

1. Related Applications

This application claims the benefit of U.S. Provisional Application Serial No. 60/125062, filed March 18, 1999, and entitled "Filtering, Scaling, and Loading of Electronic Guide Data," which is incorporated herein by reference. This application is also related generally to U.S. Patent Application Serial No. ____/____ entitled "Systems and Methods for Electronic Program Guide Data Services," which was filed on the same day as the present application, and is incorporated herein by reference.

2. The Field of the Invention

The present invention relates to electronic program guides (EPGs). More specifically, the present invention relates to application program interfaces enabling EPG data sources and applications using EPG data to interface with EPG services.

3. The Prior State of the Art

Electronic program guides (EPGs) are designed to permit television viewers as well as Internet webcast viewers to browse and select from television listings which are displayed on a television screen. Before being displayed, the EPG data is downloaded from an EPG source such as a head end or a server to a destination device such as a television, computer or set top box. The EPG source, which may receive EPG data from a number of different EPG data providers, usually consolidates and encrypts the EPG data before transmitting it to the destination device. In conventional systems, the destination device is only capable of receiving EPG data from only one EPG data source, which indicates that the destination device must accept the format of the EPG data provided by the EPG source. More

1 importantly, the EPG data received by a user cannot be tailored to that user's particular
2 tastes.

3 The ability to retrieve data from a single source in a particular format is not the only
4 drawback of this arrangement. Many destination devices which receive interactive EPG
5 data have limited storage capacity. For this reason, the EPG data downloaded from a data
6 broker or source is typically compressed such that the destination device may store more
7 EPG data in the limited storage space. This selective compression is primarily performed,
8 however, by the EPG data source. This leaves the destination device with the ability to only
9 alter the amount of EPG data downloaded as opposed to the content of the downloaded EPG
10 data.

11 In addition to being unable to alter the content of the downloaded EPG data, a
12 destination device is limited in other significant ways. EPG data from multiple EPG data
13 providers must first be collected and consolidated by a particular EPG source before it is
14 downloaded to a destination device. Destination devices are only capable of receiving EPG
15 data from a single EPG data source. Because destination devices can only receive from one
16 data provider, destination devices are limited to a single data transmission format.

17 Another limitation of current destination devices is that the applications on the
18 destination device that utilize the EPG data must be updated when the EPG data source is
19 changed or altered. Additionally, the applications on the destination device must also be
20 updated when the format of the data source is altered or changed. EPG data is currently
21 being transmitted by various EPG data sources across various transport mediums, but
22 destination devices are unable to take advantage of these sources and mediums because
23 destination devices are limited to one data provider and one format.

1 EPG data typically includes a title, a description and attributes such as closed
2 captioning, rating, and the like. EPG data can be collected from in-band and out of band
3 television signals, HTTP or FTP, TCP/IP sockets or any other communications protocol.
4 The internet, in particular, has become a significant source of EPG data. In fact, EPG data is
5 expanding to include reviews of programs, the names of the actors, and other types and
6 classes of information. Much of this information is readily available, but destination devices
7 are frequently unable to take advantage of that information because, as previously
8 mentioned, the destination devices are limited to a particular EPG data source and a
9 particular EPG data format.

10 It would be an advance in the art for a destination device to be able to receive EPG
11 data from multiple EPG data sources and not be confined to a particular EPG data source or
12 EPG data format. It would also be an advance in the art for a destination device to be able to
13 tailor the content of the EPG data by permitting the EPG data to be scaled both temporally
14 and extensibly. Scaling the EPG data in this manner provides a user, at a minimum, with
15 EPG data that is tailored to the user.

SUMMARY OF THE INVENTION

The present invention provides EPG services which allows destination devices to overcome present limitations by enabling a destination device to receive EPG data from multiple EPG data sources or providers, including web sites and other conventional EPG data providers. In order to receive EPG data from multiple EPG data sources, a software loader module is loaded or initiated for each EPG data source. The loader module is capable of retrieving EPG data from a particular EPG data source in its native transmission format. The loadable modules interface with an EPG writer module that collects the EPG data from the loader modules and stores the retrieved EPG data locally on the destination device.

The interface or EPG writer module comprises a writer application program interface (API) that establishes a standard access point for each loader module. This permits a single EPG writer module to interface with the loader modules of multiple EPG data sources. In this manner, a destination device can receive EPG data from multiple EPG data sources which have different transmission mediums and formats, because a loader module may be created or adapted to each separate EPG data source. The loader modules may receive EPG data in different formats, but all loader modules interface with the EPG writer.

Once the EPG data has been locally stored on the destination device by the EPG writer or the EPG writer API, an application may access the EPG data via a controller module which comprises a controller API. The controller API is an interface that allows many different applications to access the locally stored EPG data. The writer API and the controller API also allow applications to be independent of the EPG data sources. This independence allows an application to continue to have access to EPG data even if the EPG source or the format of the EPG data provided by the EPG source is changed.

1 EPG services also introduces significant scalability to the EPG data. Typically, the
2 EPG data stored by EPG services is for a particular time period dictated by the supplier of
3 the EPG data. EPG services allows this time period to be adjustable by a user or an
4 application. For example, a device application developer or a user may choose a time period
5 of 2 days to conserve memory on the destination device even if the EPG data source
6 provides a 5 day block of EPG data. A device application developer or a user may also
7 choose to alter the amount of data stored for titles, descriptions, and other program
8 attributes. The settings selected by a developer or an end-user can be applied to a specific
9 channel, to favorite channels, or to all channels. In other words, the EPG data stored by
10 EPG services can be customized to a particular application or user. At a minimum, a user is
11 provided with enhanced EPG data listings that can be tailored by the user by many factors
12 including but not limited to content, time, and geography.

13 The ability to tailor the content of EPG data using these attributes significantly
14 enhances how a user may make use of EPG. For example, an application developer or a
15 user may decide to download detailed information for programs occurring in the next two
16 days and to download less detailed information for programs occurring on days three
17 through five. In this example, the EPG data is temporally scaled or customized by the
18 application. In addition to being able to temporally scale or filter EPG data, EPG services
19 allows EPG data to be scaled by several factors including, but not limited to, channel,
20 richness, language, and time.

21 Additional objects and advantages of the invention will be set forth in the description
22 which follows, and in part will be obvious from the description, or may be learned by the
23 practice of the invention. The objects and advantages of the invention may be realized and
24 obtained by means of the instruments and combinations particularly pointed out in the

1 appended claims. These and other objects and features of the present invention will become
2 more fully apparent from the following description and appended claims, or may be learned
3 by the practice of the invention as set forth hereinafter.

1 **DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS**

2 Electronic program guides (EPGs) are used to convey data to a user. Primarily, the
3 data typically transmitted by an EPG data source is related to television programming and
4 may include the titles of the television programs, the descriptions of the television programs,
5 and other attributes and properties of the television programs. The EPGs are usually
6 developed by EPG data providers that are responsible for the content of the EPG data. For
7 instance, EPG data provided by an EPG data provider may be devoted exclusively to movie
8 channels or EPG data can be devoted to public or free television channels. Currently, a
9 recipient of EPG data is restricted to the format and content of a single EPG data provider.

10 EPGs, however, can also be used to convey other types of information such as radio
11 schedules, Internet streaming media event schedules, information related to the Internet, or
12 the content of the Internet. This document focuses on EPG data relating to television
13 programming, but other data that may be included in EPGs is within the scope of the present
14 invention. The television programming represented by EPG data is frequently displayed in
15 table form. Each row of the table contains a type of information such as the title of the
16 television programs, or a description of the television program, while each column is usually
17 represents a time period or time block. As described previously, a destination device is
18 currently unable to alter these characteristics and must accept the EPG data as it is formatted
19 by an EPG data source.

20 EPG services overcomes the limitations of the prior art in at least two ways. Instead
21 of simply receiving what an EPG data provider is transmitting, EPG services allows
22 incoming EPG data to be extensibly scaled according to a several different factors or axes.
23 For example, if the EPG data is temporally scaled, EPG services may receive detailed EPG
24

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17
- 18
- 19
- 20
- 21
- 22
- 23
- 24

In addition, EPG services permits a recipient of EPG data to determine the amount of descriptive EPG data such as titles, descriptions and attributes that is received. For television programs airing in the near future, the EPG data may be rich in content by including detailed descriptions while the EPG for television programs airing in the more distant future may include the titles and exclude the descriptions.

In addition to providing temporal and selective scalability, EPG services is capable of receiving EPG data from more than one EPG data source or provider. EPG services provides a loader module for each EPG data source and an abstraction layer or writer interface that interfaces with all of the loader modules. The writer interface or abstraction layer allows the EPG data received by the loader modules to be stored in the memory or storage of the destination device. Applications may then access the EPG data through another abstraction layer or control interface. The EPG data source or provider may change over time, but the application using EPG data provided by that source or provider is unaffected by those changes because the EPG data is accessed through EPG services rather than from the EPG data source or provider directly. Correspondingly, the application can be updated without regard to the EPG data provider, because the EPG data is stored via the writer interface.

The invention is described using block diagrams and flow charts. Using the block diagrams and flow charts in this manner to describe and represent the invention should not be construed as limiting the scope of the invention. Rather, the block diagrams and flow charts are intended to illustrate exemplary embodiments of the present invention.

1 The embodiments of the present invention may comprise a special purpose or
2 general purpose computer comprising various computer hardware. Embodiments within the
3 scope of the present invention also include computer-readable media having computer-
4 executable instructions or data structures stored thereon. Such computer-readable media can
5 be any available media which can be accessed by a general purpose or special purpose
6 computer. By way of example, and not limitation, such computer-readable media can
7 comprise RAM, ROM, EPROM, CD-ROM or other optical disk storage, magnetic disk
8 storage or other magnetic storage devices, or any other medium which can be used to store
9 the desired executable instructions or data structures and which can be accessed by a general
10 purpose or special purpose computer, such as one included in a set top box. Computer-
11 readable media may also include remote data stores such as web servers and other
12 remotedata stores. When information is transferred or provided over a network or other
13 communications connection to a computer, the computer properly views the connection as a
14 computer-readable medium. Thus, such a connection is also properly termed a computer-
15 readable medium. Combinations of the above should also be included within the scope of
16 computer-readable media. Computer-executable instructions comprise, for example,
17 instructions and data which cause a general purpose computer, special purpose computer, or
18 special purpose processing device to perform a certain function or group of functions. The
19 computer-executable instructions and associated data structures represent an example of
20 program code means for executing the steps of the invention disclosed herein.

21 Although not required, the invention can be described in the general context of
22 computer-executable instructions, such as program modules, being executed by one or more
23 processors included, for example, in a set top box. Generally, program modules include
24 routines, programs, objects, components, data structures, etc. that perform particular tasks or

1 implement particular abstract data types. Moreover, those skilled in the art will appreciate
2 that the invention may be practiced with other computer system configurations, including
3 general purpose computers, personal computers, hand-held devices, multi-processor
4 systems, microprocessor-based or programmable consumer electronics, network PCs,
5 minicomputers, mainframe computers, and the like.

6 As used herein "EPG data" refers to television programming information, radio
7 programming information, Internet content, scheduling information, titles, descriptions,
8 movie reviews, ratings information, MPEG Streams, and any other type of data that may be
9 retrieved from a source by an electronic device. The embodiments of the present invention
10 described herein refer to television programming information, but the embodiments of the
11 present invention easily extend to other types of EPG data. As used herein "digital
12 recording" refers to copies of recordings of the data represented by the EPG data and
13 includes but is not limited to television programming, radio programming and the like.

14 Figures 1 and 2 illustrate generally some of the advantages provided by EPG
15 services. The overall purpose of EPG services is to collect remote and transient EPG data
16 and store it to provide applications with fast access, rich content, and querying capabilities
17 for the collected EPG data. For devices having high speednetwork connectivity, the EPG
18 data may also be collected and accessed remotely. EPG services achieves these objectives
19 by: allowing the EPG data to be scaled according to many different factors or axes;
20 providing extensibility such that new sources and types of EPG data may be accessed;
21 providing components or modules to collect the EPG data from EPG data sources and
22 providers; and providing components or modules to access the EPG data that has been
23 retrieved and stored by EPG services.

1 Figure 1 is a block diagram illustrating a portion of the scalability that EPG services
2 provides for EPG data. The EPG data in Figure 1 comprises titles 106, descriptions 104 and
3 attributes 102. Titles 106, descriptions 104 and attributes 102 are each an example of EPG
4 data and the combination of titles 106, descriptions 104 and attributes 102 is also an
5 example of EPG data. EPG data may be retrieved from more than one EPG data source or
6 provider, but the EPG data that is stored by EPG services may differ from the EPG data that
7 was received from the EPG data source. Thus the amount of EPG data retrieved from a
8 particular EPG that is actually stored can vary from completely excluding the EPG data to
9 completely including the EPG data of a particular source. The EPG data illustrated in
10 Figure 1 is intended to be illustrative of EPG data in general and does not exclude additional
11 information from being included in the EPG data made available by other EPG data sources
12 and providers.

13 EPG data may be scaled, in this example, according to the factors or axes of time,
14 richness, channels and language. In Figure 1, time 108 is depicted as the horizontal axis of
15 EPG table 100. Typically, EPG data is stored for two to seven days, but time 108 can be
16 adjusted in one minute intervals to any desired interval. In addition, time 108 can be divided
17 into shorter time intervals. The richness of the EPG data refers to the amount of data that is
18 stored for television program titles, descriptions and attributes and the like. The richness of
19 the EPG data can be varied according to title, description, and attributes as well as according
20 to time 108. Thus, various intervals of time 108 may have varying levels of richness.

21 The channels is another scalable factor. In a default condition, EPG data is stored for
22 viewable channels and no EPG data is stored for hidden or test channels. A user, however,
23 may choose to remove a particular channel from the EPG data. In one embodiment, a
24 removed channel is treated differently from a test or hidden channel. EPG data is still stored

1 for removed channels, but the EPG data is not necessarily displayed to a user or an
2 application. Another aspect of the scalability of EPG services is related to favorite or
3 preferred channels. A channel that is designated as preferred or favorite may have EPG data
4 that is more rich than the EPG data stored for other channels. In addition, a user has the
5 ability to exclude certain channels or other attributes from being included in the EPG data.

6 The EPG data is also scalable with regard to language. This setting only applies,
7 however, to the storage of textual data that describes a television program. The language
8 used in the programming itself does not affect the scalability of the EPG data. The language
9 setting only permits EPG data to be stored if the language matches the system language
10 preferences. The described scalability factors of time, richness, channels and language are
11 exemplary factors and do not limit the factors with which EPG data may be scaled.

12 EPG data table 100 in Figure 1 further illustrates an example of EPG data that has
13 been scaled according to richness and time. The amount of data 109 is, in this illustration,
14 indicative of the richness of the EPG data. Column 110 contains attributes 102, descriptions
15 104 and titles 106. The time period for column 110 is, in this example, the next two days of
16 EPG data. The EPG data in column 110 is rich in that descriptions 104 are extremely
17 detailed, titles 106 are complete, and many attributes 102 are detailed. Exemplary attributes
18 include closed captioning, rating, stereo and more.

19 Column 111 is indicative, in this example, of a time period from day 3 to day 5. As
20 illustrated, the amount of data 109 is less and is coarser or less rich than the EPG data stored
21 in column 110. Descriptions 104 in column 111 are not as detailed, but titles 106 are still
22 complete and column 111 contains no attributes 102. Similarly, column 112 is indicative of
23 a time period from day 6 to day 7. Only titles 106 are stored in column 112. Thus the EPG
24 data stored in column 112 is less rich than the EPG data in column 111. EPG data table 100

1 therefore illustrates EPG data that has been scaled with respect to time and richness or
2 content.

3 The following table is another illustration of EPG data that has been scaled according
4 to time, richness, and channel. The EPG data displayed to a user will reflect the EPG data
5 indicated in the following table.

<u>Channel Type</u>	<u>EPG Data</u>	<u>Time</u>
Channels Hidden by EPG Data Provider	Title Description Attributes	Do not store Do not store Do not store
Normal Channels	Title Description Attributes	Store 4 days Store 3 days Store 2 days
Favorite Channels	Title Description Attributes	Store 5 days Store 5 days Store 5 days

12 The above table illustrates a scenario where no EPG data is stored for hidden
13 channels, the richness of the EPG data stored for normal channels is greater for television
14 programs airing in the near future, and the EPG data stored for favorite channels is very rich
15 for the entire time period stored. In this manner, the EPG data can be scaled and stored in a
16 variety of different combinations using a variety of factors. EPG services allows EPG data
17 to be custom tailored for a particular application or user. Channels or other attributes or
18 descriptions that are not of interest to a particular application or user may be excluded.

19 Figure 2 is a block diagram illustrating an exemplary system capable of
20 implementing one embodiment of the present invention. Figure 2 illustrates a device 20,
21 which may be a set top box, a computer, a video cassette recorder, a television tuner, a
22 modem, a digital satellite tuner, or any other device capable of receiving EPG data. Device
23 20 receives EPG data from EPG data provider 70. Exemplary EPG data providers or
24 sources include television stations, Internet Service Providers (ISPs), and satellites. The

1 EPG data may be transferred via in-band television signals, out-of-band television signals,
2 Hyper Text Transfer Protocol (HTTP), File Transfer Protocol (FTP), Transmission Control
3 Protocol/Internet Protocol (TCP/IP) sockets, or any other communication protocol.

4 Device 20 typically has at least one application 22 which is capable of utilizing the
5 EPG data provided by EPG data provider 70. EPG services 40 acts as an intermediary
6 between EPG data provider 70 and application 22. In this manner, both application 22 and
7 EPG data provider 70 may change independent of the other without creating an
8 incompatibility that may render application 22 incapable of receiving EPG data because
9 EPG data provider 70 and application 22 interact with EPG services 40 rather than with each
10 other.

11 One embodiment of EPG services 40 provides pluggable modules for EGP loaders
12 60, EPG writer 43, storage 42, and EPG control 41. EPG loaders 60 provide the ability to
13 receive EPG data from EPG data providers. As new EPG data providers become available
14 to device 20, a new EPG loader 60 is added to EPG services 40. The EPG loader added to
15 the system for the new EPG data provider is capable of collecting the EPG data from the
16 new EPG data provider. For example, an EPG loader 60 may be added to EPG services 40
17 which is capable of retrieving or collecting movie review data from an HTTP address. In
18 addition, EGP loaders 60 can be added and removed from EPG services 40 as needed.
19 When an EPG data provider 70 is no longer available, for example, the EPG loader
20 corresponding to that EPG provider is removed from EPG services 40.

21 EPG writer 43 is a component or module that stores the EPG data collected by EPG
22 loaders 60 in storage 42 which is a database containing EPG data in one embodiment.
23 Storage 42 is also an example of a computer-readable medium. EPG writer 43 is
24 responsible, in this example, for enforcing the scaling of the EPG data as well as the

1 maximum amount of storage 42 that may be used for storing EPG data. Thus, if an EPG
2 loader attempts to store EPG data that is greater in size than the maximum amount of EPG
3 data allowed, then EPG data relating to television programming furthest in the future is
4 removed until the EPG data being stored does not exceed the maximum storage amount
5 allowed.

6 EPG writer 43 also removes EPG data from storage 42 that is no longer needed or
7 usable. For instance, EPG writer 43 deletes the EPG data from storage 42 for all television
8 programs which have aired before the current half-hour block. For example, at 8:15 a.m.,
9 EPG writer 43 will delete all EPG data relating to television programming occurring prior to
10 8:00 a.m. At 8:35 a.m., EPG writer 43 deletes television programming prior to 8:30 a.m.
11 One reason for deleting EPG data and for limiting the amount of EPG data that can be
12 downloaded is because storage 42 of device 20 is typically limited.

13 Because EPG services 40 may have a plurality of EPG loaders 60, it is possible that
14 one or more EPG loaders 60 will attempt to add EPG data to the same record or storage
15 locations. More simply, it is possible for a conflict to arise with respect to the EPG data in a
16 variety of contexts. For this reason, EPG services provides various schemes for resolving
17 conflicts. For example, multiple loaders may attempt to update the description for the same
18 program. To resolve this conflict, the EPG writer may be implemented to treat each EPG
19 loader with a different priority. The EPG writer may permit higher priority loaders to take
20 precedence. Another alternative for EPG loaders with equal priorities is to permit the last
21 EPG data written to take precedence and be stored in storage 42. Another alternative is to
22 allow a user or application to select a conflict resolution scheme if conflicting EPG data
23 sources are present. Simply stated, the user could designate a particular EPG data source or
24

1 provider as having priority over another IPG data source or provider. These are examples of
2 conflict resolution, but other methods or schemes may be employed.

3 It is also possible for EPG loaders 60 to implement a priority scheme independent of
4 the EPG writer. This implementation, however, reduces the independence of the EPG loaders
5 and forces more rigid interface and implementation constraints upon the development and
6 design of the EPG loaders. In addition, EPG loaders 60 may be time based threads that
7 activate and deactivate at predetermined times.

8 The retrieval of EPG data from storage 42 is performed by a control module
9 illustrated as EPG control 41, which is an example of a controller API in some
10 embodiments. EPG control 41 may access: EPG data related to the available channels
11 including channel number, channel station, network identification, station broadcast times
12 and associated Uniform Resource Locators (URLs); EPG data related to information related
13 to programming information including title, description, length, rating and other air times; or
14 EPG data relating to whether the television program is purchasable, such as a pay-per-view
15 movie or event. The EPG data can be accessed and retrieved by EPG control 41 in a variety
16 of different combinations.

17 The ability to access and retrieve EPG data is enhanced by the querying capabilities
18 supplied by EPG control. EPG data can be searched by many parameters, including but not
19 limited to, time, title, description, category or other attribute which may be user defined. In
20 this manner, the EPG data retrieved by an application can be extremely customized for an
21 application. For high performance applications, EPG control 41 is also capable of notifying
22 application 22 when the EPG data within a particular range changes. This enables
23 applications 22 to maintain up to date EPG data available.
24

1 Figure 3 is a detailed block diagram of a system implementing one embodiment of
2 the present invention. Device 20 has one or more applications 22 which are capable of using
3 EPG data. The EPG data retrieved by applications 22 may be displayed to a user via a
4 television screen or a computer screen, or may be sent to a printer or other rendering device.
5 EPG data is typically generated by EPG data providers 70, which are shown in Figure 3 as
6 listings provider 74, movie descriptions provider 75 and satellite listings provider 76. In this
7 example, the EPG data generated by listings provider 74 is transmitted via a television
8 station 71, the EPG data generated by movie descriptions provider 75 is transmitted via an
9 Internet Service Provider (ISP) 72, and the EPG data generated by satellite listings provider
10 76 is transmitted via satellite 73. It is understood that EPG data can be transmitted in a
11 variety of different methods and that the transmission of EPG data described herein is
12 exemplary rather than limiting. For instance, listings provider 74 may also transmit via ISP
13 72.

14 The EPG data provided by EPG data providers 70 is received at device 20 by EPG
15 services 40 which comprises EPG loaders 60. An EPG loader or loader module has the
16 responsibility of collecting EPG data from its native transmission format. EPG loaders 60
17 comprises one or more specific EPG loaders because as illustrated by the various EPG data
18 providers 70, EPG data can be transmitted in a variety of different methods and formats and
19 new sources or providers of EPG data may become available in the future. For these and
20 other reasons, each loader module collects EPG data, in this embodiment, from a particular
21 source or provider.

22 Figure 3 illustrates a Vertical Blanking Interval (VBI) loader module 61, a HTTP
23 loader module 62 and a MPEG loader module 63. VBI loader module 61 is capable of
24 collecting EPG data from the VBI of a television signal or broadcast. HTTP loader module

62 is capable of collecting EPG data over a network such as the Internet, and MPEG loader module 63 is capable of collecting or retrieving EPG data from a digital stream provided by satellite 73. EPG loaders 60 allow EPG data to be collected from any communication system or protocol and are not limited to those illustrated in Figure 3.

In addition to understanding the transmission medium and format of a particular EPG data provider 70, Each EPG loader module optionally comprises the hardware and software necessary to accomplish this task. This is illustrated by HTTP loader module 62 which comprises a protocol 64, a device driver 65 and a modem 66. For HTTP loader module 62, protocol 64 may be HTTP and TCP/IP if the transmission medium is the Internet. Typically, modem 66 is the hardware required to access the Internet, but access can be accomplished with other hardware components such as a network interface cards. Device driver 65 is typically a software module designed for a particular hardware component such as modem 66.

Conflict resolution 44 is an optional pluggable component that permits a user to specify the actions to be taken when one or more EPG data providers 70 conflict. The action taken by conflict resolution 44 can be specified by the user, the application 22 or the supplier of the EPG data. For instance, the user could specify that conflict resolution 44 cause movie descriptions provider 75 to have priority over satellite listing sprovider 76. In this instance, the portions of the EPG data provided by satellite listings provider 76 that conflict with the EPG data provided by movie descriptions provider 75 are discarded or otherwise handled. Conflict resolution 44 is extensible and can be implemented by attribute as well as by EPG data provider. Conflict resolution 44 can also be implemented at the EPG loader level, whereby the EPG loaders resolve conflicts.

1 Each EPG loader 60 is capable of interacting with EPG writer 43. EPG writer 43
2 comprises an applications programming interface (API), in one embodiment, which is used
3 by EPG loaders 60 to interface with EPG writer 43. The collected EPG data is placed in
4 storage 42 by EPG writer 43. Typically, EPG loader 60 creates an instance of EPG writer
5 43 to perform updates to storage 42, which may be a local or remote data store. EPG writer
6 43 provides methods having one or more parameters that allow the EPG data collected by
7 EPG loaders 60 to be placed in storage 42 such that the EPG data may be queried and
8 retrieved by EPG control 41.

9 EPG writer 43 provides the following properties, methods or functions.
10 AudioSubChannel adds a new audio subchannel format and has the parameters
11 languageCode, isMainAudioService, isDolbyEncoded, and programType.
12 AudioSubChannelForScheduleEntry adds a new audio subchannel to a schedule entry and
13 has the parameters scheduleDataID and audioDataID. CategorizationSystem adds a new
14 categorization system to the data store and has the parameters categorizationSystemName
15 and pSystemDataID. CategoryPair adds a new category or subcategory pair to the data
16 store.

17 CategoryForProgram maps a category pair to a specific program and has the
18 parameters programDataID and categoryPairDataID. Channel adds a new channel to the
19 data store and has the parameters serviceRecordID, channel, network, station, description,
20 channelTtype, startSeconds, and pChannelDataID. ClearAll removes all data from the data
21 store and has no parameters. ClearChannels removes all channel data from the data store
22 and has no parameters. ClearPrograms removes all program data from the data store and has
23 no parameters. ClearScheduleEntries removes all schedule data from the data store.
24

1 DeleteAudioSubChannel removes a specific audio subchannel from the data store and has
2 the parameter audioDataID.

3 DeleteCategorizationSystem removes a specific categorization system from the data
4 store and has the parameter systemDataID. DeleteCategoryPair removes a specific category
5 pair from the data store and has the parameters systemDataID, categoryName, and
6 subCategoryName. DeleteChannel removes a specific channel from the data store had has
7 the parameter ChannelObject. DeleteProgram removes a specific program from the data
8 store and has the parameter ProgramObject. DeletePropertyForChannel removes a property
9 from a channel object and has the parameters pChannelObject and propertyName.
10 DeletePropertyForProgram removes a property from a program object and has the
11 parameters pProgramObject and propertyName.

12 DeletePropertyForScheduleEntry removes a property from a schedule entry and has
13 the parameters pScheduleEntryObject and propertyName. DeletePropertyForWeblink
14 removes a property from a weblink object and has the parameters pWeblinkObject and
15 propertyName. DeleteScheduleEntry removes a specific schedule entry from the data store
16 and has the parameter ScheduleEntryObject. DeleteSharingDay removes a specific sharing
17 day from the data store and has the parameter sharingDayDataID. DeleteVideoSubChannel
18 removes a specific video subchannel from the data store and has the parameter videoDataID.
19 DeleteWeblink removes a specific Web link from the data store and has the parameter
20 WeblinkObject.

21 EnableDuplicateChecking sets the preferred result for this condition and has the
22 parameter preventDuplicates. Program adds a new program to the data store and has the
23 parameters programTitle, programDescription, yearMade, seconds, and programDataID.
24 PropertyForChannel adds an extensible name-value property to a specific channel and has

1 the parameters channelDataID, name and value. PropertyForProgram adds an extensible
2 name-value property to a specific program and has the parameters programDataID, name,
3 and value. PropertyForScheduleEntry adds an extensible name-value property to a specific
4 schedule entry and has the parameters scheduleEntryDataID, name, and value.
5 PropertyForWeblink adds an extensible name-value property to a specific Weblink and has
6 the parameters weblinkDataID, name, and value.

7 PurchaseStringForScheduleEntry adds a new purchase string to a specific schedule
8 entry and has the parameters scheduleEntryDataID and purchaseString, which is frequently
9 encrypted. RatingForProgram maps a rating authority and rating to a specific program and
10 has the parameters program DataID, and ratingCode.
11 RemoveOldScheduleEntriesAndPrograms removes schedule entries before a specific time
12 and their associated program, rating, category, property, and Web link data and has the
13 parameter beforeThisTime. ScheduleEntry adds a new schedule entry to the data store and
14 has the parameters eventID, channelDataID, startTime, endTime, cc, rerun, videoDataID,
15 audioDataID, and scheduleEntryDataID. SharingDay adds a unique sharing day to the data
16 store and has the parameters startDay, startTime, endDay, endTime, and pSharingDataID.
17 SharingDayForChannel links a unique sharing day to a particular channel and has the
18 parameters channelDataID, and sharingDataID.

19 UpdateComplete indicates the current set of updates to the data store is complete.
20 VideoSubChannel indicates the current set of updates to the EPG services storage or data
21 store is completer and has the parameters language, aspectRation, videoSourceHeight,
22 videoSourceWidth, squarePixel, componentType, streamed, and pVideoDataID.
23 VideoSubChannelForScheduleEntry creates a relationship between a schedule entry and a
24 video subchannel and ahs the parameters scheduleEntryDataID and videoDataID.

1 Weblink adds a new Web link to the data store and has the parameters url,
2 description, startTime, endTime and pWeblinkDataID. WeblinkForChannel maps a
3 Weblink to a specific channel and has the parameters channelDataID, and weblinkDataID.
4 WeblinkForProgram maps a Web link to a specific program and has the parameters
5 programDataID, and weblinkDataID. WeblinkForScheduleEntry maps a Web link to a
6 specific schedule entry and has the parameters scheduleEntryDataID and weblinkDataID.

7 While EPG loaders 60 place EPG data in storage 42 via EPG writer 43, EPG control
8 41 performs the functions necessary to retrieve the EPG data from storage 42. EPG control
9 41, in one embodiment also comprises an API that provides functions and methods to
10 application 22 that assist in the access and retrieval of EPG data. For example, it is
11 necessary in many instances for EPG control 41 to determine if the necessary EPG data is
12 available in storage 42. Functions and methods are also provided by EPG control 41 that are
13 able to access and retrieve EPG data relates to channels.

14 While a program is typically a unique event, programs may be repeated by a
15 television station and may therefore correspond to one or more schedule entries. To retrieve
16 EPG data concerning a program, EPG control 41 provides methods and functions for use
17 with schedule entries. Programs may also belong to one or more categories or subcategories
18 and EPG control 41 provides functions to retrieve the valid category and subcategory pairs.

19 Programs also may have one or more ratings which correspond to different rating
20 authorities and EPG control 41 has functions and methods to access ratings. Some
21 applications 22 may request notification of changes to the EPG data that may occur within a
22 particular time range. EPG control 41 services these requests and fires an event when the
23 EPG data changes. Typically, only one event may be requested for each instance of EPG
24 control 41. EPG control 41 provides properties, methods, and events, generally designated

1 as functions, which allow EPG control 41 to be used by an application to access storage 42
2 or other data store.

3 The properties of EPG control 41 are as follows. CategorizationSystems retrieves a
4 collection of the names of known schemes for organizing programs by type and has the
5 parameter psystems. DataEndTime returns the time in the future at which the available data
6 ends and has the parameter pEndTime. HighestDataStartTime returns the furthest time in
7 the future when a program starts and has the parameter pStartTime. IsAnyDataAvailable
8 returns information to confirm if valid channel or guide listings exist in the data store and
9 has the parameter pDataAvailable. IsChannelDataAvailable returns information to indicate
10 if channel data exists in the data store and has the parameter pChannelDataAvailable.
11 RatingSystem retrieves a collection of strings for the names of known schemes or systems
12 for organizing content ratings and has the parameter psystems. AvailableChannels retrieves
13 the Channels collection object and has the parameter pChannels.

14 The methods of EPG control 41 are as follows. CancelCategoryEventRequest
15 disables signaling of OnCategoryUpdate events. CancelChannelEventRequest disables
16 signaling of OnChannelUpdate events. CancelRangeEventRequest disables signaling of
17 OnRangeUpdate events. Categories retrieves a collection of the names of main categories
18 within a categorization system for a given categorization system name and has the
19 parameters categorizationSystemName and pCategories.
20 ChannelsForProviderNetworkName returns channels matching the search value and has the
21 parameters searchString, substringMatch, and pChannels. ChannelsForNumber retrieves a
22 Channels collection object for valid device channels with a particular channel number or
23 tuple and has the parameters serviceSpace, channel, time, and pChannels.
24

1 ChannelsForProviderName returns channels matching the search value and has the
2 parameters searchString, substringMatch, and pChannels.

3 IsScheduleDataAvailable returns information indicating whether EPG data is found
4 for a range and has the parameters startTime, endTime, and pScheduleDataAvailable.
5 Program retrieves the Program object representing the program shown on the specified
6 channel at the specified time and has the parameters serviceSpace, channel, time and
7 pProgram. ProgramEndTime retrieves the end time for the program shown on the specified
8 channel at the specified time and has the parameters serviceSpace, channel, time and
9 pEndTime.

10 ProgramLength retrieves the length in seconds for the program shown and the
11 specified channel and the specified time and has the parameters serviceSpace, channel, time
12 , and pSeconds. ProgramRating retrieves a Rating object for a particular program and has
13 the parameters serviceSpace, channel, time, ratingSystem, and prating. ProgramStartTime
14 retrieves the start time for a particular program shown on the specified channel at the
15 specified time and has the parameters serviceSpace, channel, time, and pStartTime.
16 ProgramTitle retrieves the name of the program shown on the specified channel at the
17 specified time and has the parameters serviceSpace, channel, time, and pTitle.

18 RequestCategoryUpdateEvent indicates that an event should be fired when any
19 aspect of the known categorization systems changes. RequestChannelUpdateEvent indicates
20 that an event should be fired when a new channel has been added to the specified service
21 space. RequestRangeUpdateEvent returns the update of the time range and has the
22 parameters startTime and endTime. ScheduleEntries returns all schedule time slots
23 matching the query values and has the parameters serviceSpace, channel, startTime,
24 endTime, and pScheduleEntries. ScheduleEntriesForCategories returns programs that match

1 the query values and has the parameters categorizationSystem, categories, startTime,
2 endTime, maxHits, and pScheduleEntries.

3 ScheduleEntriesForProgramProperty returns programs matching the query values
4 and has the parameters name, value, startTime, endTime, maxHits, and pScheduleEntries.

5 ScheduleEntriesForProperty returns time slots matching query values and has the parameters
6 name, value, startTime, endTime, maxHits, and pScheduleEntries.

7 ScheduleEntriesForStrings retrieves a ScheduleEntries collection object representing all
8 time periods for programs whose title or description includes a particular case-insensitive
9 string which will be shown and has the parameters searchStrings, subStringMatch, findTitle,
10 findDescription, startTime, endTime, maxHits, and pScheduleEntries. ScheduleEntry
11 returns a time slot matching query values, and has the parameters serviceSpace, channel,
12 time, and pScheduleEntry. SubCategories retrieves a collection of the names of
13 subcategories within a categorization system for a given categorization system name and
14 category name and has the parameters categorizationSystemName, categoryIndex, and
15 pSubCategories.

16 The EPG control 41 events are as follows. OnCategoryUpdate indicates that a new
17 category has been added. OnChannelUpdate indicates that a new channel has been added.
18 OnRangeUpdate indicates that the data within the time range specified by
19 RequestRangeUpdateEvent has changed.

20 The extensible properties of a channel, program, schedule entry, weblink or other
21 EPG data is retrieved in one embodiment through the use of EPG control objects which
22 include CategoryPair, CategoryPairs, Channel, Channels, GuideRating, GuideRatings,
23 Program, Programs, PurchaseStrings, ScheduleEntry, ScheduleEntries, Weblink, Weblinks,
24 SharingSchedule, MajorMinorSelectorIDs, and Tuple. These objects and associated

1 methods are detailed in the previously incorporated provisional application. While the APIs
2 of the invention have been described herein with reference to specific functions and
3 parameters, the invention also extends to other APIs that provide similar functionality.

4 EPG services is intended to provide extensibility to EPG data. As illustrated, new
5 sources of EPG data are easily accommodated by a device. The EPG data can be scaled and
6 formatted in a variety of ways. Extensible properties can be added to the channel, program
7 schedule entry, or weblink data and multiple properties may be added to the same set of
8 EPG data, which enhances the ability of applications to search and utilize the stored EPG
9 data.

10 EPG Data Map is another component of EPG services that is used to retrieve
11 necessary data from EPG services, but should not be used by EPG loaders or other
12 applications or components. EPG Data Map, in one embodiment, is an API.

13 EPG Data Map provides the following methods and functions. NetworkCallLetters
14 retrieves the network name for a specific service record ID and has the parameters
15 serviceRecordID, time, and pNetworkCallLetters. ProgramEndTime retrieves the end time
16 for the program shown on the specified channel at the specified time and has the parameters
17 serviceRecordID, time, and pEndTime. ProgramLength retrieves the length in seconds for
18 the program shown on the specified channel at the specified time and has the parameters
19 serviceRecordID, time, and pSeconds. ProgramPurchaseString retrieves the available
20 purchase strings for a specified program and has the parameters serviceRecordID, time,
21 cDims, pcDims, and pPurchaseStrings. ProgramRatings retrieves Ratings objects for the
22 program shown on the specified channel at the specified time and has the parameters
23 serviceRecordID, time, and pRatings. ProgramStartTime retrieves the start time for the
24 program shown on the specified channel at the specified time and has the parameters

1 serviceRecordID, time, and pStartTime. ProgramTitle retrieves the name of the program
2 shown on the specified channel at the specified time and has the parameters
3 serviceRecordID, time, and pTitle. ServiceRecordIDForNetworkCallLetters retrieves a
4 ServiceRecordID that matches a specific network call letter string and has the parameters
5 serviceSpace, searchString, substringMatch, cDims, pcDims, and pServiceRecordIDs.
6 ServiceRecordIDForStationCallLetters retrieves a ServiceRecordID that matches a specific
7 station call letter string and has the parameters serviceSpace, searchString, substringMatch,
8 cDims, pcDims, and pServiceRecordIDs. StationCallLetters retrieves the station name for a
9 specific service record ID and has the parameters serviceRecordID, time, minorChannel, and
10 pStationCallLetters.

11 The present invention may be embodied in other specific forms without departing
12 from its spirit or essential characteristics. The described embodiments are to be considered
13 in all respects only as illustrative and not restrictive. The scope of the invention is,
14 therefore, indicated by the appended claims rather than by the foregoing description. All
15 changes which come within the meaning and range of equivalency of the claims are to be
16 embraced within their scope.

17 What is claimed and desired to be secured by United States Letters Patent is:
18
19
20
21
22
23
24

1 1. In a device having one or more electronic program guide (EPG) loaders
2 enabling the device to receive EPG data from one or more EPG sources, a method for
3 interfacing the one or more EPG loaders with a database associated with the device, the
4 method comprising the steps of:

5 calling a function of a writer module operating at the device that collects the
6 EPG data from the EPG loaders and that stores the EPG data in the database; and
7 executing the function by the writer module.
8

9 2. A method as in claim 1, wherein the function comprises AudioSubChannel
10 having parameters comprising languageCode, isMainAudioService, isDolbyEncoded, and
11 programType.
12

13 3. A method as in claim 1, wherein the function comprises
14 AudioSubChannelForScheduleEntry having parameters comprising scheduleDataID and
15 audioDataID.
16

17 4. A method as in claim 1, wherein the function comprises
18 CategorizationSystems having parameters comprising categorizationSystemName and
19 systemDataID.
20

21 5. A method as in claim 1, wherein the function comprises CategoryPair having
22 parameters comprising systemDataID, categoryName, subCategoryName, and
23 categorypairDataID.
24

14. A method as in claim 1, wherein the function comprises DeleteCategoryPair having parameters comprising systemDataID, categoryName, and subCategoryName.

1 15. A method as in claim 1, wherein the function comprises DeleteChannel
2 having parameters comprising channelObject and propertyName.

3
4 16. A method as in claim 1, wherein the function comprises Delete
5 PropertyForProgram having parameters comprising programObject and propertyName.

6
7 17. A method as in claim 1, wherein the function comprises
8 DeletePropertyForScheduleEntry having parameters comprising scheduleEntryObject and
9 propertyName.

10
11 18. A method as in claim 1, wherein the function comprises
12 DeletePropertyForWeblink having parameters comprising weblinkObject and
13 propertyName.

14
15 19. A method as in claim 1, wherein the function comprises DeleteScheduleEntry
16 having parameter scheduleEntryObject.

17
18 20. A method as in claim 1, wherein the function comprises DeleteSharingDay
19 having parameter sharingDayDataID.

20
21 21. A method as in claim 1, wherein the function comprises
22 DeleteVideoSubChannel having parameter videoDataID.

1 22. A method as in claim 1, wherein the function comprises DeleteWeblink
2 having parameter weblinkObject.

3
4 23. A method as in claim 1, wherein the function comprises
5 EnableDuplicateChecking having parameter preventDuplicates.

6
7 24. A method as in claim 1, wherein the function comprises Program having
8 parameters comprising programTitle, programDescription, yearMade, seconds, and
9 programDataID.

10
11 25. A method as in claim 1, wherein the function comprises PropertyForChannel
12 having parameters comprising channelDataID, name, and value.

13
14 26. A method as in claim 1, wherein the function comprises PropertyForProgram
15 having parameters comprising programDataID, name, and value.

16
17 27. A method as in claim 1, wherein the function comprises
18 PropertyForScheduleEntry having parameters comprising scheduleEntryDataID, name, and
19 value.

20
21 28. A method as in claim 1, wherein the function comprises PropertyForWeblink
22 having parameters comprising weblinkDataID, name, and value.

1 29. A method as in claim 1, wherein the function comprises
2 PurchaseStringForScheduleEntry having parameters comprising scheduleEntryDataID and
3 purchaseString.
4

5 30. A method as in claim 1, wherein the function comprises RatingForProgram
6 having parameters comprising programDataID and ratingCode.
7

8 31. A method as in claim 1, wherein the function comprises
9 RemoveOldScheduleEntriesAndPrograms having parameter beforeThisTime.
10

11 32. A method as in claim 1, wherein the function comprises ScheduleEntry
12 having parameters comprising eventID, channelDataID, startTime, endTime, cc, rerun,
13 videoDataID, audioDataID, and scheduleEntryDataID.
14

15 33. A method as in claim 1, wherein the function comprises SharingDay having
16 parameters comprising startDay, startTime, endDay, endTime, and pSharingDataID.
17

18 34. A method as in claim 1, wherein the function comprises
19 SharingDayForChannel having parameters comprising channelDataID, and sharingDataID.
20

21 35. A method as in claim 1, wherein the function comprises UpdateComplete.
22
23
24

1 36. A method as in claim 1, wherein the function comprises VideoSubChannel
2 having parameters comprising language, aspectRation, videoSourceHeight,
3 videoSourceWidth, squarePixel, componentType, streamed, and videoDataID.

4
5 37. A method as in claim 1, wherein the function comprises
6 VideoSubChannelForScheduleEntry having parameters comprising scheduleEntryDataID,
7 and videoDataID.

8
9 38. A method as in claim 1, wherein the function comprises Weblink having
10 parameters comprising url, description, startTime, endTime, and pweblinkDataID.

11
12 39. A method as in claim 1, wherein the function comprises WeblinkForChannel
13 having parameters comprising channelDataID, and weblinkDataID.

14
15 40. A method as in claim 1, wherein the function comprises WeblinkForProgram
16 having parameters comprising programDataID, and weblinkDataID.

17
18 41. A method as in claim 1, wherein the function comprises
19 WeblinkForScheduleEntry having parameters comprising scheduleEntryDataID, and
20 weblinkDataID.

21
22 42. A computer-readable medium having computer-executable instructions for
23 performing the steps recited in claim 1.

24

1 43. In a device having electronic program guide (EPG) data from one or more
2 EPG data sources stored in a database associated with the device, a method for interfacing
3 the EPG data stored in the database with one or more applications operating at the device,
4 the method comprising the steps of:

5 calling a function of a control module operating at the device that accesses
6 the database to retrieve the EPG data and transmits the EPG data to the one or more
7 applications; and

8 executing the function by the control module.

9
10 44. A method as in claim 43, wherein the function comprises
11 CategorizationSystems having parameter psystems.

12
13 45. A method as in claim 43, wherein the function comprises DataEndTime
14 having parameter endTime.

15
16 46. A method as in claim 43, wherein the function comprises
17 HighestDataStartTime having parameter startTime.

18
19 47. A method as in claim 43, wherein the function comprises
20 IsAnyDataAvailable having parameter dataAvailable.

21
22 48. A method as in claim 43, wherein the function comprises
23 IsChannelDataAvailable having parameter channelDataAvailable.

1 49. A method as in claim 43, wherein the function comprises RatingSystems
2 having parameter systems.

3
4 50. A method as in claim 43, wherein the function comprises AvailableChannels
5 having parameter channels.

6
7 51. A method as in claim 43, wherein the function comprises
8 CancelCategoryEventRequest.

9
10 52. A method as in claim 43, wherein the function comprises
11 CancelChannelEventRequest.

12
13 53. A method as in claim 43, wherein the function comprises
14 CancelRangeEventRequest.

15
16 54. A method as in claim 43, wherein the function comprises Categories having
17 parameters comprising categorizationSystemName and categories.

18
19 55. A method as in claim 43, wherein the function comprises
20 ChannelsForProviderNetworkName having parameters comprising searchString,
21 substringMatch and channels.

1 56. A method as in claim 43, wherein the function comprises
2 ChannelsForNumber having parameters comprising serviceSpace, channel, time and
3 channels.

4
5 57. A method as in claim 43, wherein the function comprises
6 ChannelsForProviderName having parameters comprising searchString, substringMatch,
7 and channels.

8
9 58. A method as in claim 43, wherein the function comprises
10 IsScheduleDataAvailable having parameters comprising startTime, endTime and
11 scheduleDataAvailable.

12
13 59. A method as in claim 43, wherein the function comprises Program having
14 parameters comprising serviceSpace, channel, time and program.

15
16 60. A method as in claim 43, wherein the function comprises ProgramEndTime
17 having parameters comprising serviceSpace, channel, time, and endTime.

18
19 61. A method as in claim 43, wherein the function comprises ProgramLength
20 having parameters comprising serviceSpace, channel, time, and seconds.

21
22 62. A method as in claim 43, wherein the function comprises ProgramRating
23 having parameters comprising serviceSpace, channel, time, ratingSystem, and rating.

63. A method as in claim 43, wherein the function comprises ProgramStartTime having parameters comprising serviceSpace, channel, time, and startTime.

64. A method as in claim 43, wherein the function comprises ProgramTitle having parameters comprising serviceSpace, channel, time, and title.

65. A method as in claim 43, wherein the function comprises RequestCategoryUpdateEvent.

66. A method as in claim 43, wherein the function comprises RequestChannelUpdateEvent.

67. A method as in claim 43, wherein the function comprises RequestRangeUpdateEvent having parameters comprising startTime, and endTime.

68. A method as in claim 43, wherein the function comprises ScheduleEntries having parameters comprising serviceSpace, channel, startTime, endTime, and scheduleEntries.

69. A method as in claim 43, wherein the function comprises ScheduleEntriesForCategories having parameters comprising categorizationSystem, categories, startTime, endTime, maxHits, and scheduleEntries.

1 70. A method as in claim 43, wherein the function comprises
2 ScheduleEntriesForProgramProperty having parameters comprising name, value, startTime,
3 endTime, maxHits, and scheduleEntries.

4
5 71. A method as in claim 43, wherein the function comprises
6 ScheduleEntriesForProperty having parameters comprising name, value, startTime,
7 endTime, maxHits, and scheduleEntries.

8
9 72. A method as in claim 43, wherein the function comprises
10 ScheduleEntriesForStrings having parameters comprising searchString, substringMatch,
11 findTitle, findDescription, startTime, endTime, maxHits, and scheduleEntries.

12
13 73. A method as in claim 43, wherein the function comprises ScheduleEntry
14 having parameters comprising serviceSpace, channel, time, and scheduleEntry.

15
16 74. A method as in claim 43, wherein the function comprises SubCategories
17 having parameters comprising categorizationSystemName, categoryIndex, and
18 subCategories.

19
20 75. A method as in claim 43, wherein the function comprises OnCategoryUpdate.

21
22 76. A method as in claim 43, wherein the function comprises OnChannelUpdate.

23
24 77. A method as in claim 43, wherein the function comprises OnRangeUpdate.

1 78. A computer readable medium having computer executable instructions for
2 performing the steps recited in claim 43.
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

WORKMAN, NYDEGGER & SEELEY

A PROFESSIONAL CORPORATION

ATTORNEYS AT LAW

1000 EAGLE GATE TOWER

60 EAST SOUTH TEMPLE

SALT LAKE CITY, UTAH 84111

ABSTRACT OF THE INVENTION

EPG services are provided, which enables a device such as a computer, a set top box or the like to collect EPG data from multiple EPG data providers or sources. The EPG data is collected at the device by EPG loaders. Each EPG loader collects EPG data from a particular source. The EPG loaders deliver the EPG data to a writer module, which performs scaling and formatting functions to the EPG data. The writer module also stores the scaled and formatted EPG data in a database or other storage. The writer module may also be implemented by chaining EPG loaders in a priority scheme. However, EPG loader interaction reduces independence and forces more rigid interface and implementation constraints upon the design of EPG loaders. Once stored, applications gain access to the EPG data through a control module. The control module provides the accessed EPG data to the applications. EPG services function to isolate the operation of the applications from the EPG data sources. Because the EPG data is stored by the writer module and accessed by the control module, the applications and EPG data sources can change independently of one another without having an effect on the ability of EPG services to collect and provide access to EPG data.

G:\DATA\PAT\WORDPAT\14531572.DOC

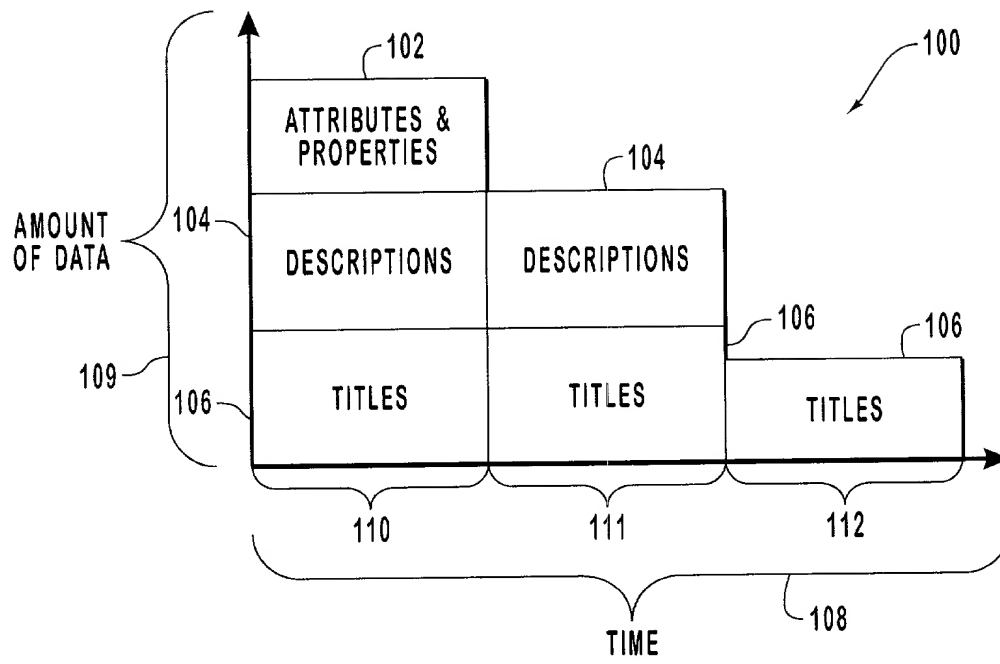


FIG. 1

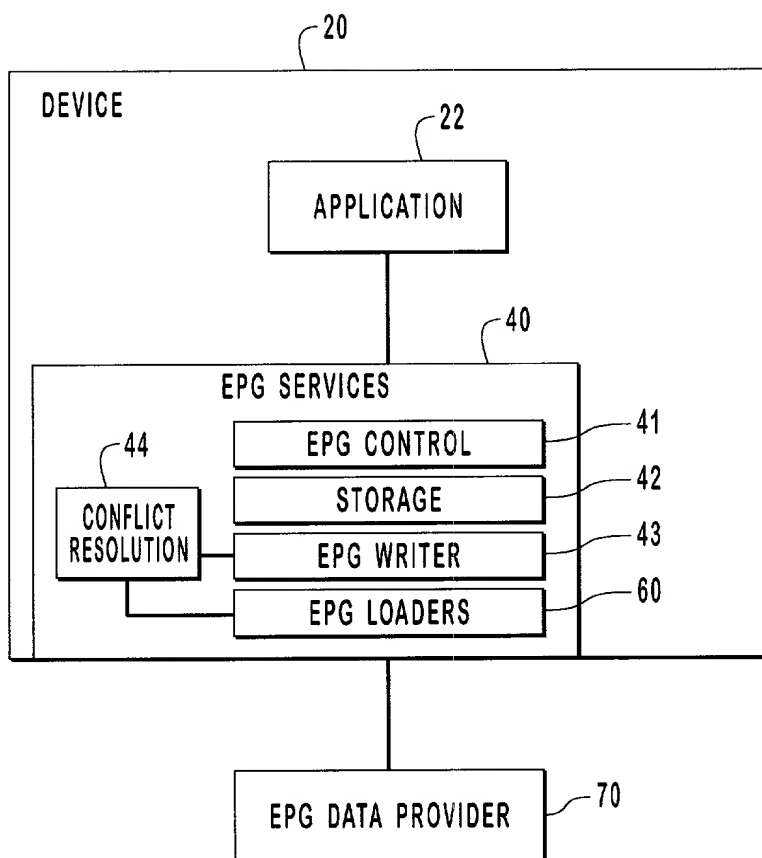


FIG. 2

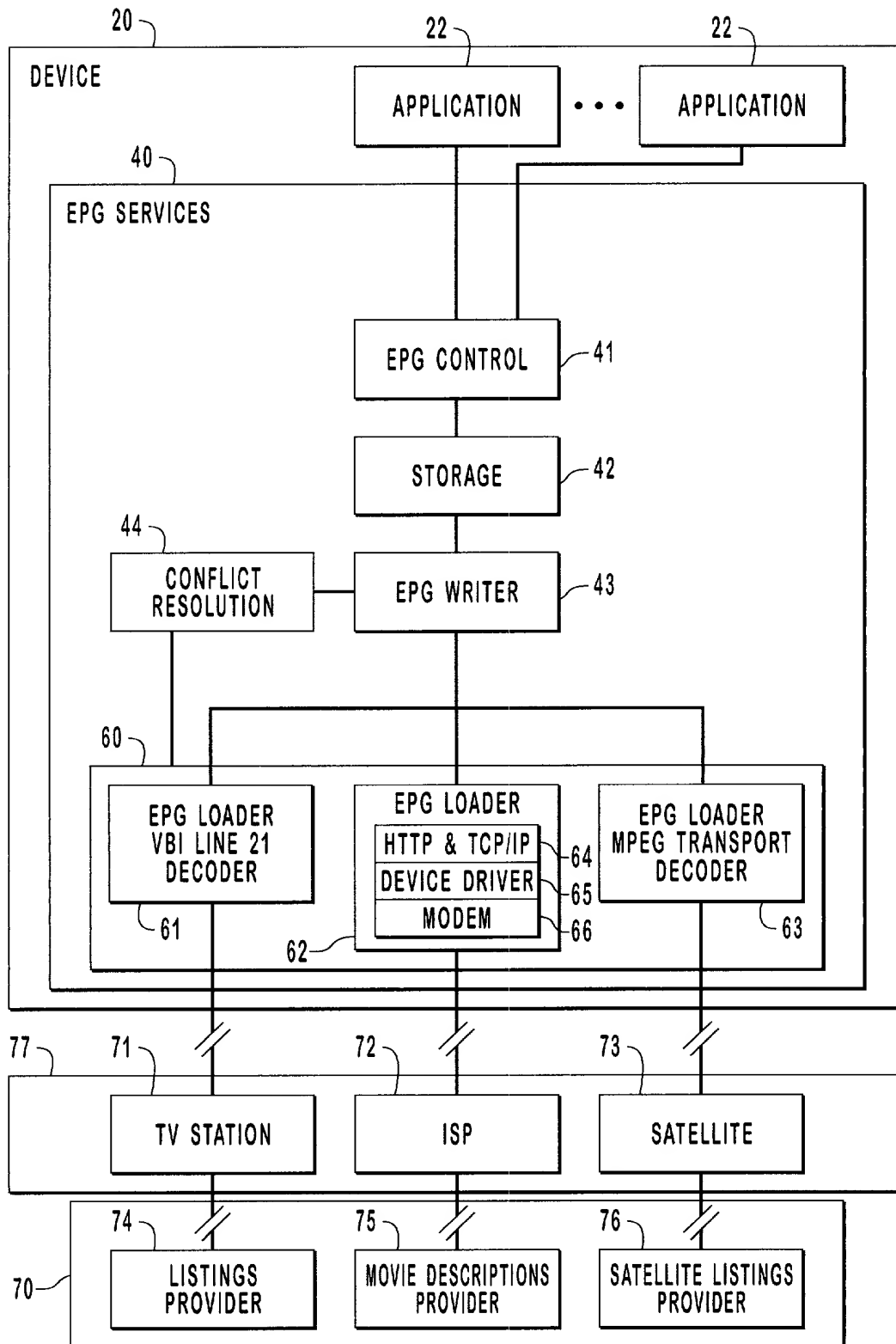


FIG. 3